In the Claims

1. (currently amended) A computer implemented method for solving a combinatorial optimization problem including a plurality of elements and a plurality of values, comprising:

applying a priority algorithm in a form of an ordering function to an instance of the combinatorial optimization problem to produce an ordering of the elements;

modifying the ordering of the elements to produce a re-ordering of the elements;

applying a placement function to map values to the corresponding elements of the re-ordering; and

repeating the modifying and the applying until all elements have been placed to obtain a solution of the combinatorial optimization problem.

2. (currently amended) The method of claim 1, in which the priority algorithm is fixed.

3. (currently amended) The method of claim 1, in which the priority algorithm is dynamic.

4. (original) The method of claim 1, in which the re-ordering is within a predetermined distance of the ordering.

5. (original) The method of claim 4, in which the distance is a Kendall-tau distance.

6. (currently amended) The method of claim 1, in which the re-ordering uses a decision vector, and in which the ~~distance~~ decision vector has one field for each element of the order, each field determining a new order of the element in the re-ordering.

7. (original) The method of claim 1, in which the re-ordering is probabilistic.

8. (new) A computer program product storing a computer program which when executed by a computer performs a method for solving a combinatorial optimization problem including a plurality of elements and a plurality of values by performing the steps of:

applying a priority algorithm in a form of an ordering function to an instance of the combinatorial optimization problem to produce an ordering of the elements;

modifying the ordering of the elements to produce a re-ordering of the elements;

applying a placement function to map values to the corresponding elements of the re-ordering; and

repeating the modifying and the applying until all elements have been placed to obtain a solution of the combinatorial optimization problem.

9. (new) A computer implemented method for solving a combinatorial optimization problem including a plurality of elements and a plurality of values, comprising:

applying a priority algorithm in a form of an ordering function to an instance of the combinatorial optimization problem to produce an ordering of the elements, in which the priority algorithm is dynamic;

modifying the ordering of the elements to produce a re-ordering of the elements;

applying a placement function to map values to the corresponding elements of the re-ordering; and

repeating the modifying and the applying until all elements have been placed to obtain a solution of the combinatorial optimization problem.

10. (new) The method of claim 9, in which the re-ordering is within a predetermined distance of the ordering.

11. (new) The method of claim 10, in which the distance is a Kendall-tau distance.

12. (new) The method of claim 9, in which the re-ordering uses a decision vector, and in which the decision vector has one field for each element of the order, each field determining a new order of the element in the re-ordering.

13. (new) The method of claim 9, in which the re-ordering is probabilistic.